Manual for WAND² Event Filter and Sequential Fit

July 2021
Daria Soboleva, Virtual Summer Research Intern
at Oak Ridge National Laboratory
Mentor: Matthias D. Frontzek

OAK RIDGE National Laboratory | HIGH FLUX ISOTOPE REACTOR

Index

<mark>I) Event Filtering</mark>:

WAND^2 is recording data in event mode. This means each neutron gets its individual timestamp. In this section, we explain how to filter the data for neutrons with a certain timestamp (or any recorded parameter; for instance, temperature).

1) Go to **File** on the left upper corner and hit "**Open Script**" or **Ctrl+O.**
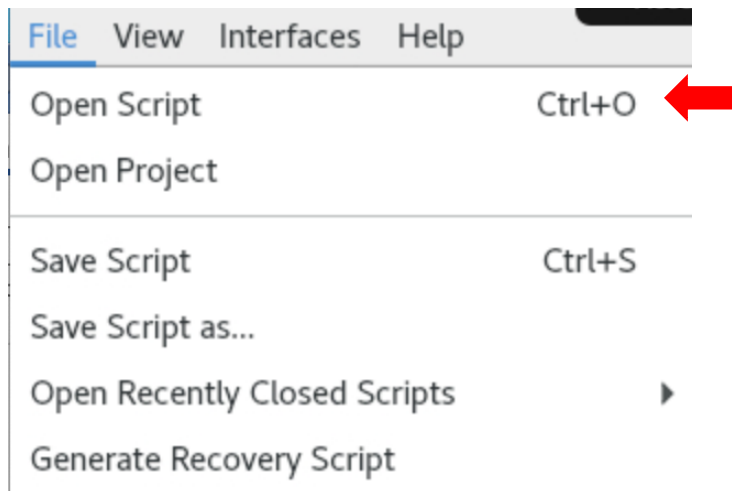
Follow the path data→HFIR→HB2C→shared→WANDscripts and locate "**Filter_combined_ZYP.py**".

wand2_scd_reduction.py				4.4 kB	18 Feb
New					20 Feb
writer.py				2.5 kB	7 Mar
writer2.py				2.5 kB	7 Mar
Old					9 Mar
workaround_nuclear.py				1.1 kB	12 Apr
save_multiple_powder.py				1.1 kB	19 Apr
SCD_MDE_Vana.py				1.4 kB	29 Apr
scd					24 May
Filter_combined_ZYP.py				6.9 kB	25 May
mnf					2 Jun

2) After the script has been loaded, go to the first paragraph of the script and fill out all the information: name of the output workspace, IPTS and Run Numbers for both your data and Vanadium (background), NormaliseBy, Units, Binning, and Sample Log that will be used to filter.

<mark>Back to Index</mark>

```
19 # ----------------------------- Input section ------------------------------
20 ##########################################################################
21
22 # Inputs for runs information.
23 title = 'output_th'
24 IPTS = 7776
25 run = 327629
26 background = None
27 BackgroundScale = 1
28 vanadium = 531821   # Run number or `None`
29 vanadium_IPTS = 23858
30
31 # Inputs for runs control.
32 normaliseBy = 'Monitor'   # One on (None, Monitor, Time)
33 units = 'Theta'   # One of (Theta, ElasticQ, ElasticDSpacing)
34 Binning = '5,125,1200'   # Min,Max,Number_of_bins
35 log_binning = False
36
```

Name your output file

Fill out the IPTS and Run Numbers for both the sample and Vanadium

In the second paragraph, modify "normaliseBy", "units", and "Binning" if needed.

3) You can filter by time by typing "**True**" **OR** you can filter by Sample Log (only one option can be picked at once).

```
37 # Inputs concerning event filter. First, choose to filter by time or sample log,
38 # depending on the value of `filter_by_time`.
39 #
40 # !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
41 # N.B. The option to filter by time and sample log is NOT compatible with each othe
42 # !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
43 #
44 filter_by_time = False
45
46 # Inputs concerning event filter by time.
47 if filter_by_time:
48     start_time = '0'
49     stop_time = '100'
50     time_interval = '10'
51     unit_of_time = 'Seconds'
52     time_tolerance = 0
53
54 # Inputs concerning event filter by sample log.
55 LogName = 'HB2C:SE:SampleTemp'
56 LogValueInterval = 25
57 MinimumLogValue = None
58 MaximumLogValue = None
59 log_val_tolerance = LogValueInterval / 2.0
60 log_change_direction = 'Both'
61
```

Change "False" to "**True**" to filter by time and modify the next paragraph as necessary.

If using Sample Log, go to "Show Sample Logs" and choose the Sample Log. To find the Sample Log, use LoadWAND to upload the data, then right click on the workspace created and select "**Show Sample Logs**".

Back to Index

If filtering by Sample Log is chosen, it is not recommended to have more than 20 intervals for a reasonable time of processing (e. i. if the temperature runs from 100K to 600K, the maximum value for the interval would be 25K).

4) When finished, hit Run.

5) Once the script has finished running, the new output workspace (the one that was created in the previous step – default name is "output_th") can be accessed in the "Workspace" window. Right clicking on the first file to plot individually. You can also select to plot spectrum with error depending on your goal.

6) Right click on another file and select "**Overplot spectrum**".
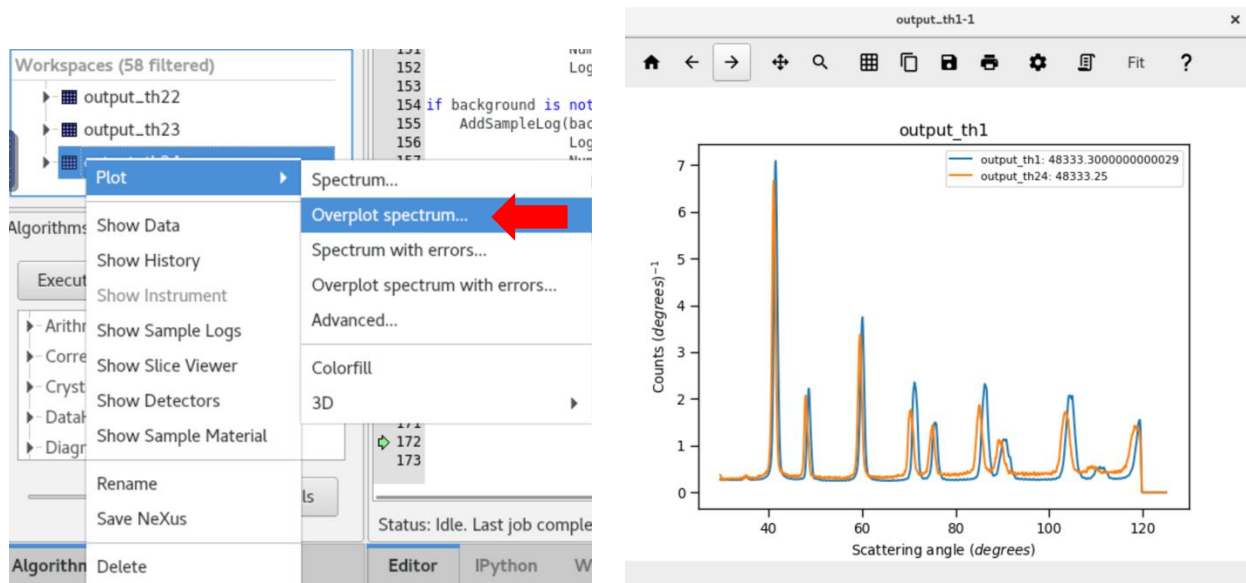
Click on the zoom to view an individual peak shift.
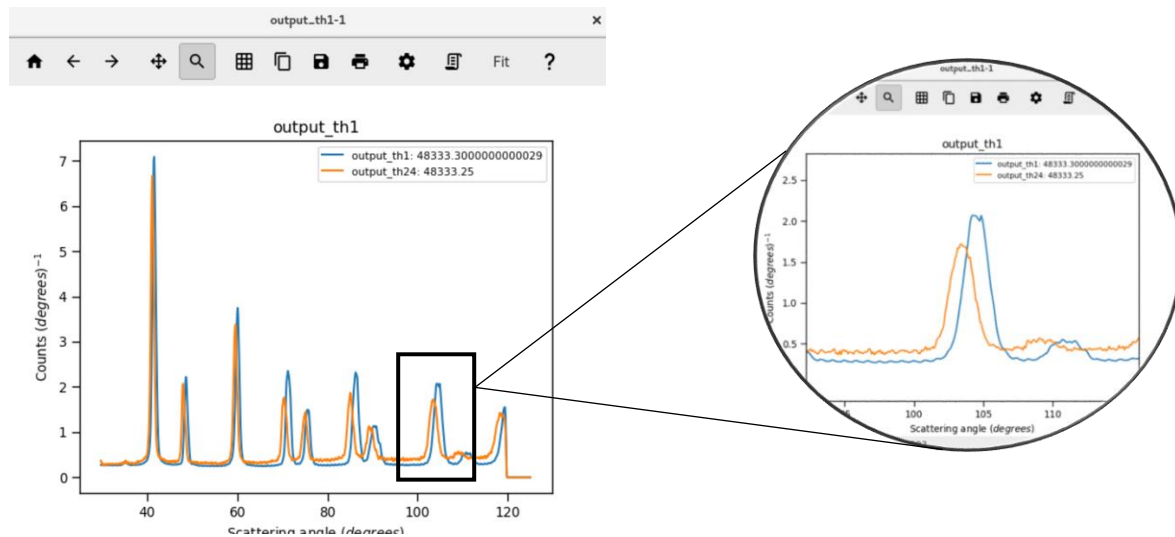
II) Fitting the peaks.

1) In this section, we will explain how to extract parameters from multiple data sets from event filtering. In order to fit the peaks, plot the folder ("**Spectrum**" or "**Spectrum with error**") and click on "**Fit**" on top a plot.
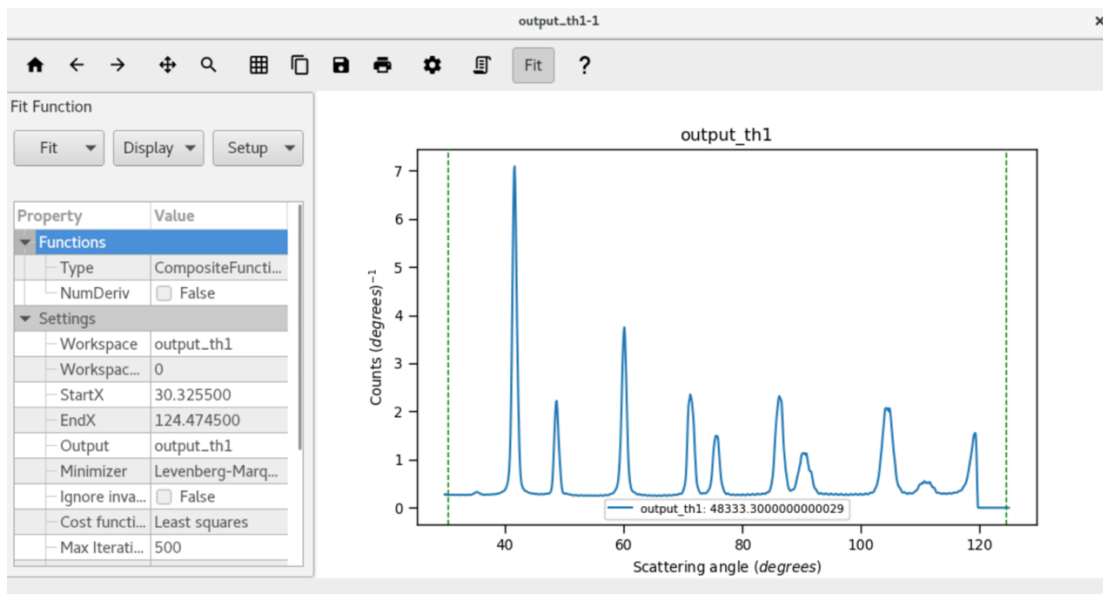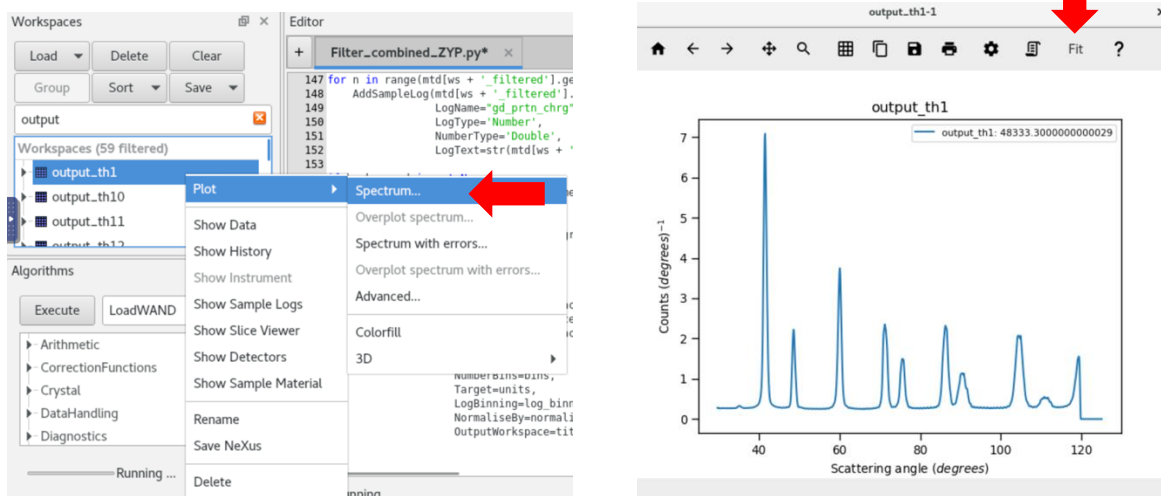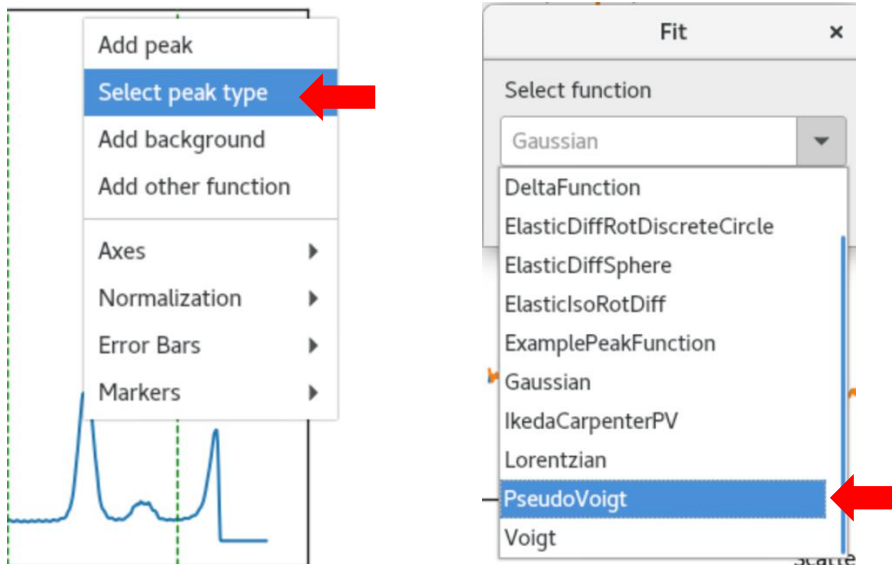
2) Move the green lines or enter the interval for fitting manually.
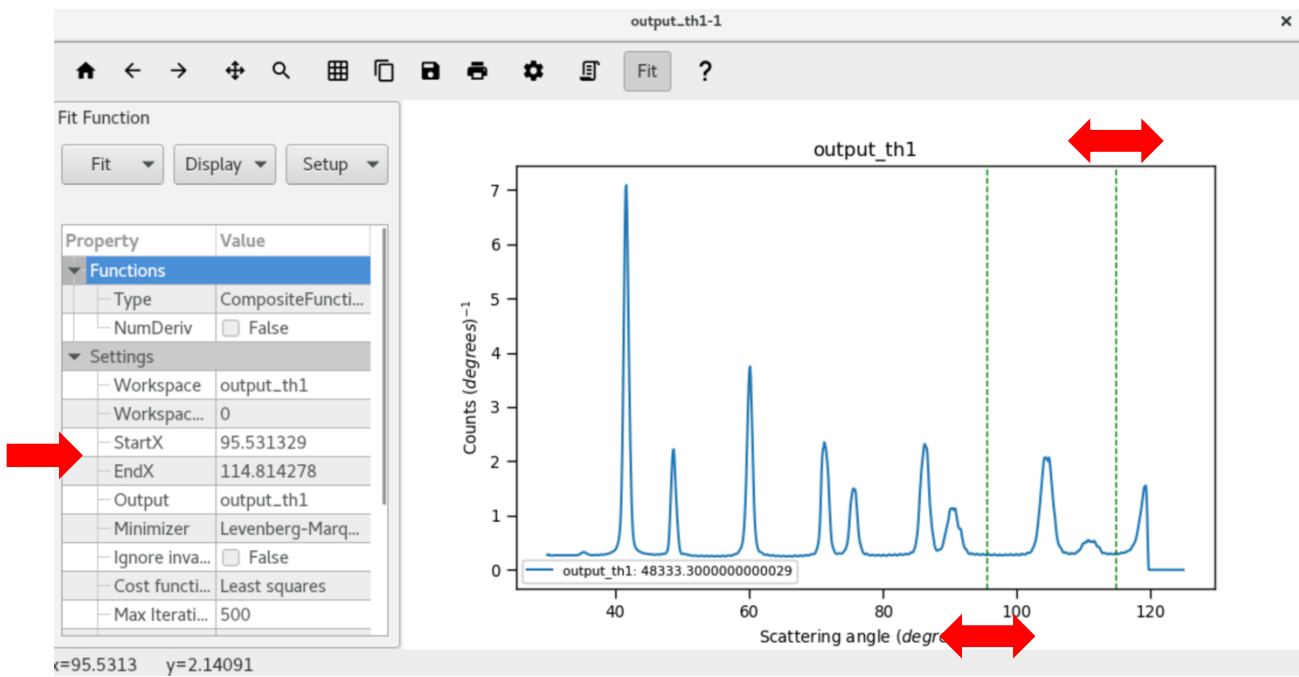
3) Right click on the peak to "**Add background**".

Linear Background is sufficient.
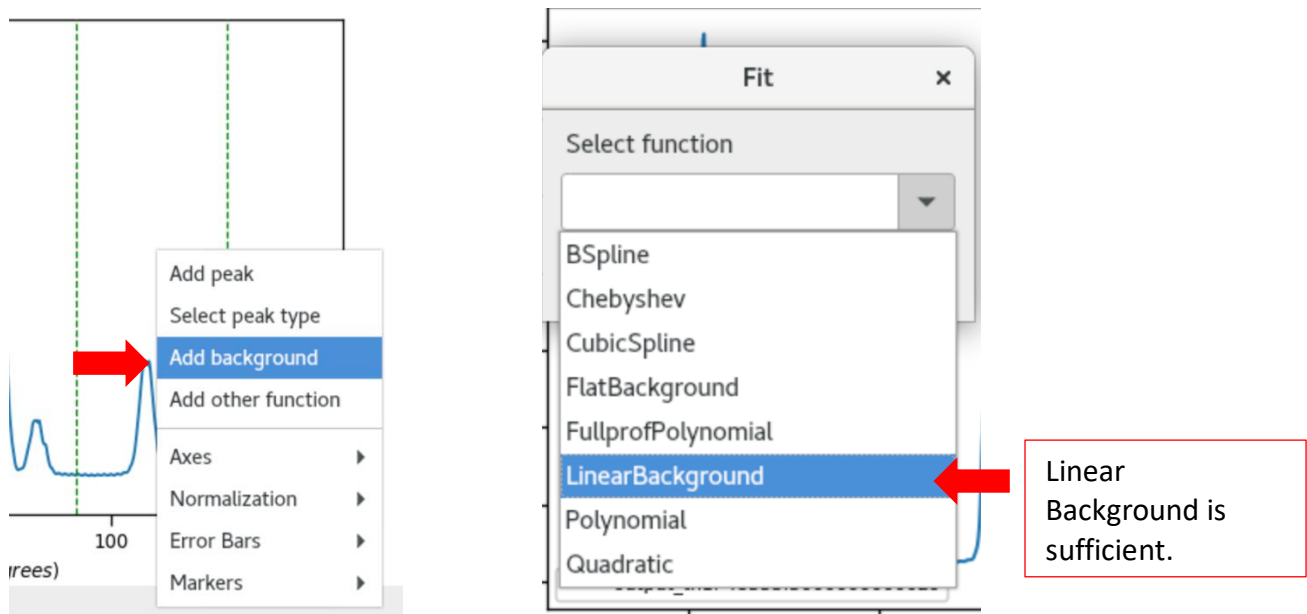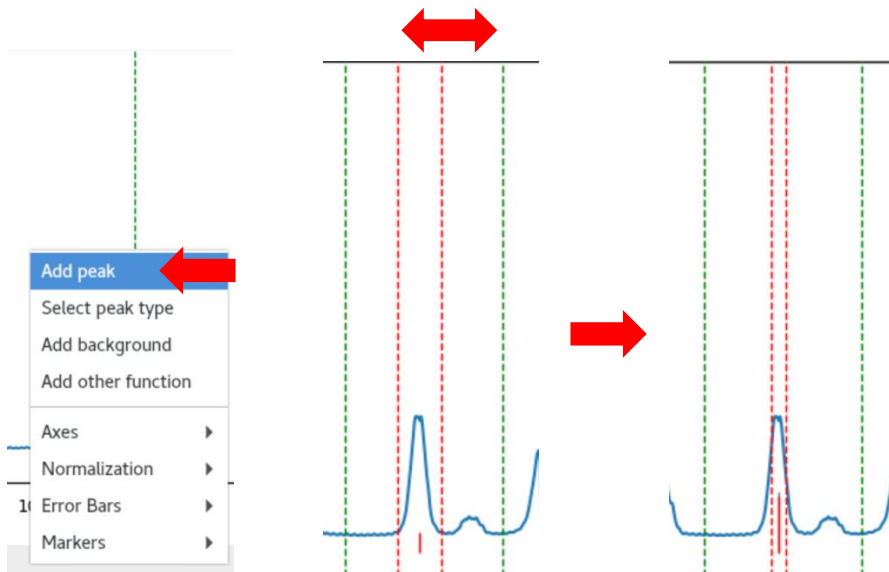
4) Right click on the peak to "**Select peak type**".

"**PseudoVoigt**" is the general choice, based on the instrument resolution. If your sample has significant sample peak broadening due to microstructure or for other reasons, a different peak shape may be necessary.

5) Right click and select "**Add peak**", click on the peak to fit and adjust the red lines to fit the peak by moving them.

6) There are two ways to fit the peak: you can either fit it individually or use sequential fit.

      a. To fit, click on "**Fit**" and then select "**Fit**".
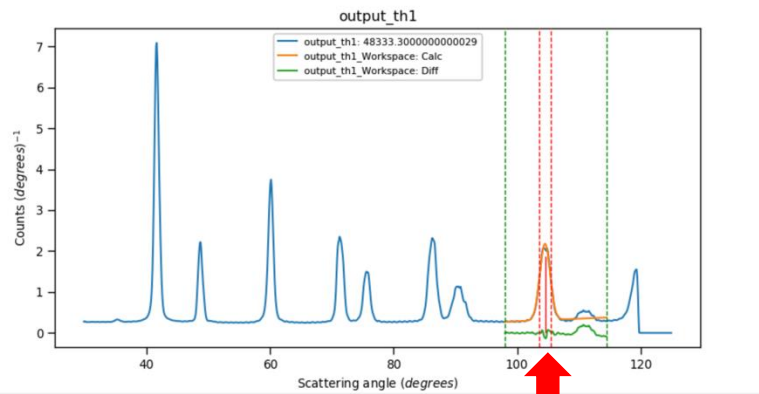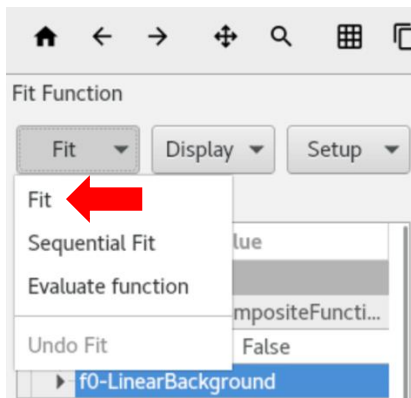
The graph shows the fitted pick and the residual, which is the difference between the fitted peak and the data.

b. For sequential fitting to fit multiple patterns sequentially using the same dialogue:

i. After that, click on "**Fit**" and then on "**Sequential Fit**".

"Parameter" and "Curve Type" spaces should not be modified because they are used to create an output plot, which is currently not implemented.

It is necessary modify the Log Value depending on the Sample Log that was used on for data collection and in the script.

Back to Index

ii. Click on "**Add Workspaces**".
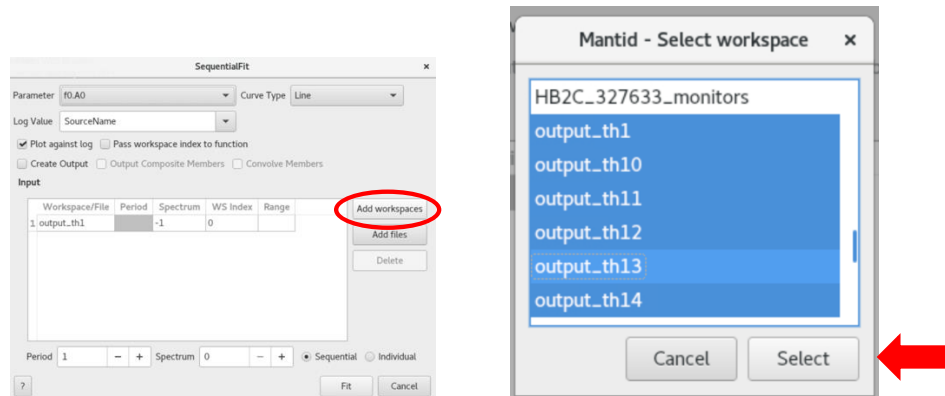
iii. After all the workspaces added, there are two options for fitting:

- Sequential fit – the parameters of a previous fit are used for every new fit.
- Individual fit – the parameters of the initial fit are used for every new fit.
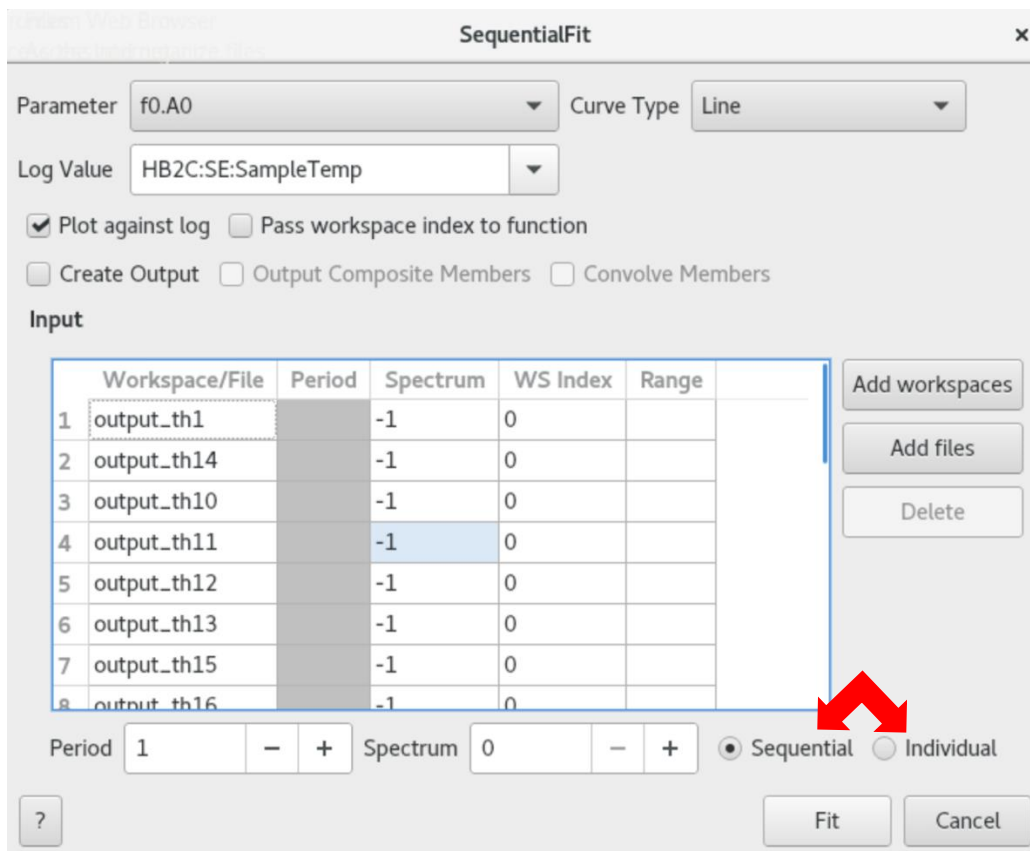
iv. After all the adjustments are done, click "**Fit**".

7) In order to view, go to the workspaces and find the file "**Name_of_your_file_res**". Right click on it and select "**Show data**".

8) To plot it, select a column, right click on it and select an option. The Sample Log values are automatically selected as x-values.

9) To save the data, go to Algorithms and execute "**SaveASCII**" to save the table in a comma separated file.

Back to Index
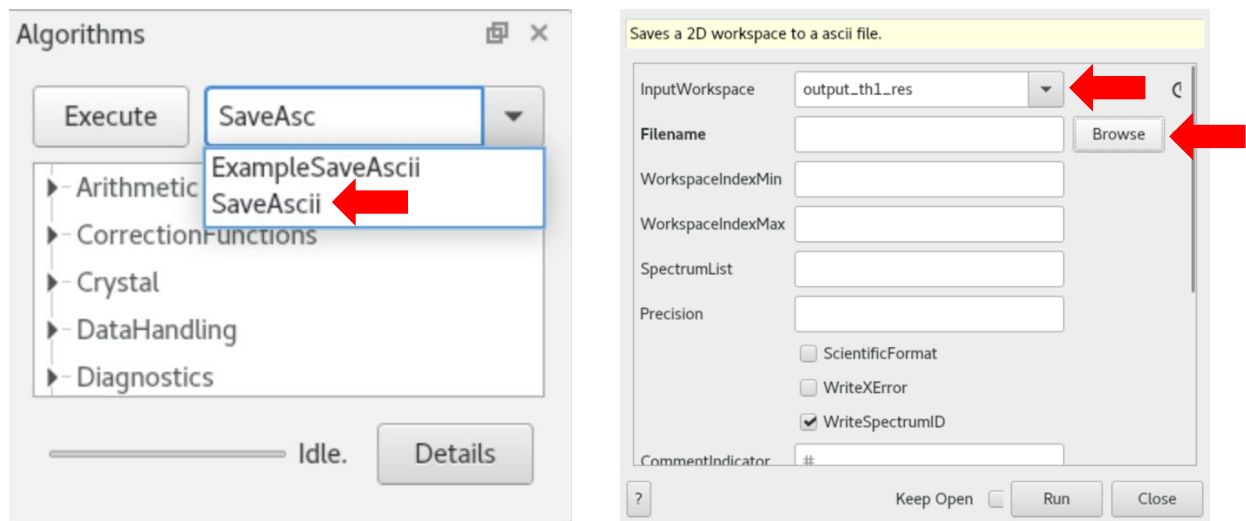
Type in the "**Name_of_your_file_res**" as the Input Workspace. Then click on Browse, name the file and click on Save.
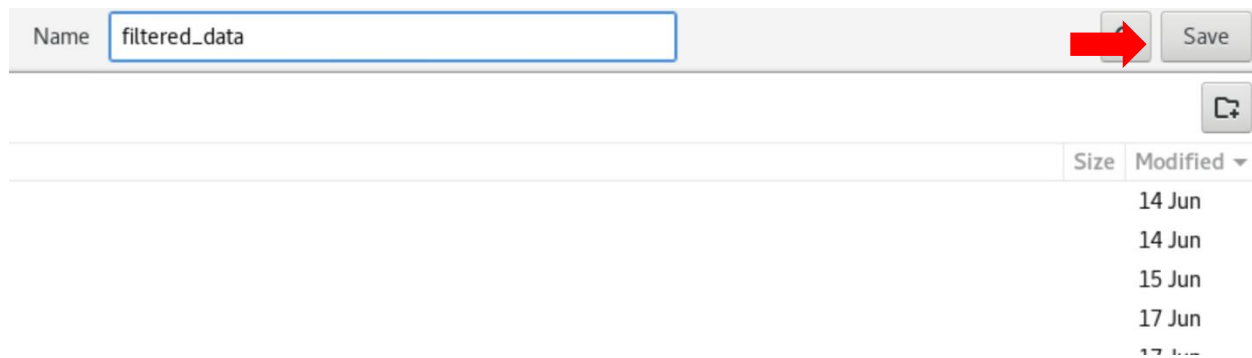
10) If you want to download these files for use with Rietveld refinements on your local PC/laptop, you can either use a server storage service like Google drive or Dropbox, or download using FileZilla or WinSCP (link on the webpage).

Back to Index